TRUCK ALLOCATION MODEL USING LINEAR PROGRAMMING AND QUEUEING THEORY

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF

Bachelor of Technology

In

Mining Engineering

By

Soubhagya Sahoo

(Roll No.: 108MN011)

Department of Mining Engineering

National Institute of Technology, Rourkela

Odisha

April, 2012

TRUCK ALLOCATION MODEL USING LINEAR PROGRAMMING AND QUEUEING THEORY

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF

Bachelor of Technology

In

Mining Engineering

By

Soubhagya Sahoo

(Roll No.: 108MN011)

Under the guidance of

Dr. B. K. Pal

Department of Mining Engineering

National Institute of Technology, Rourkela

Odisha

April, 2012

National Institute of Technology

Rourkela

CERTIFCATE

This is to certify that the thesis entitled, "**TRUCK ALLOCATION MODEL USING LINEAR PROGRAMMING AND QUEUEING THEORY**" submitted by Soubhagya Sahoo in partial fulfilment of the requirements for the award of Bachelor of Technology Degree in Mining Engineering at the National Institute of Technology, Rourkela (Deemed University) is an authentic work carried out by them under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university/institute for the award of any Degree or Diploma.

Date:

Dr. B. K. Pal

Dept. of Mining Engineering

National Institute of Technology, Rourkela

Rourkela-769008

Acknowledgement

The research work related to this thesis has been carried out at Department of Mining Engineering, National Institute of Technology (NIT), Rourkela. This work would have been impossible without the help and guidance of several people, whose contribution I would like to acknowledge.

First of all, I would like to express our sincere and profound gratitude to our supervisor Dr. B.K. Pal, Professor of Mining Engineering, Department of Mining Engineering, National Institute of Technology, Rourkela, for his consistent encouragement and tremendous support throughout our research work and sharing his technical knowledge has led us through many difficulties easily. This thesis work would have been a difficult task to complete without profiting from his expertise, encouragement and valuable time and criticisms. His endless drive for new and better results is highly appreciated.

Finally, I would like to express our heart-felt gratitude to our parents and our family members for being with us when encountering difficulties. Their loving support has been and always will be our most precious possession on earth.

Place: National Institute of Technology, Rourkela

Date: Soubhagya Sahoo

Abstract

Truck-Shovel is the most common means used for transportation of ore/waste in surface mining operations, but it is the most costly unit procedure in a truck-shovel mining system. The present advancement in computing technology offers the potential of refining truck-shovel productivity and consequent savings. Introducing a truck allocation model in a mine can attain operational improvements by decreasing waiting times and achieve other benefits through enhanced optimal routing and grade control. The efficiency of the working truck-shovel fleet is determined by the allocation model in use, the complexity of the truck shovel system and a multiplicity of other variables. In maximum cases computer simulation is the most applicable and operational method of comparing the different allocation approaches.

A model is presented here to minimize the number of trucks allocated to a set of shovels, considering throughput and ore grade constraints. A nonlinear relation between a shovel's throughput and the number of trucks allocated to the shovel using queueing theory and linear programming is being established. It is assumed that each shovel is allocated a single truck size. Different linear programming methods are being suggested to optimize the constraints.

Table of Contents

List of Figures and Tables

Acronyms and Abbreviations

M	Exponential distribution (Markovian)
D	Degenerate distribution (constant times)
E_k	Erlang distribution (shape parameter = k)
G	General distribution (any arbitrary distribution allowed
f(x)	Objective function value
x_q	Non-basic variable
µ	Mean service rate
x_j	Basic variable
S	Set of shovels
L	Lower limit on throughput
g_l	Lower limit on ore grade
g_u	Upper limit on ore grade
t	Number of trucks
s_s^2	Squared coefficient of variation of shovel
s_a^2	Squared coefficient of variation of truck
W	Shovel throughput
C	Truck capacity
$P_0(t)$	Shovel idle probability
$1/\lambda$	Mean of back-cycle time distribution
ρ	Utilization factor
z	Number of trucks
x_k	Binary variable
x_{kj}	Binary linearization variable

z_j	Integer variable for the number of trucks allocated to shovel j
Wj	Continuous variable for the throughput of shovel j

CHAPTER-1

Introduction

1.1 Introduction

Surface mining comprise the elementary actions of overburden removal, drilling and blasting, mineral loading, hauling and dumping and numerous different secondary processes. Loading of ore and waste is administered concurrently at many different locations within the pit and infrequently in many different pits. Shovels and frond-end loaders of assorted sizes are used for loading material to trucks. Haul roads are often extraordinarily complicated, cover giant surface areas and undergo extreme elevation changes. The shovel loading time depends on shovel capacity, digging conditions, and the capacity of the truck. At the shovels queues are formed since various sizes of trucks may be used at individual shovels. Thus, the allocation of trucks to haul specific material from a specific pit makes it a complicated problem. Certainly, efficient mining operations strongly depend on proper allocation of truck-shovel and the corresponding allocation of trucks along the dump sites. The two important factors in determining the optimum design parameters of an open-pit mining system are the number and type of trucks and shovels. The characteristics of truck's entrance and loading times at shovels govern the performance actions (i.e. total production) of truck-shovel system. The expectations of identical truck travel and loading times may result in misjudging or miscalculating the performance of these processes.

It is very useful to measure the performance of a truck-shovel system in open-pit mines precisely. Any bordering upgrading in the performance would save a important amount of money in most current open-pit mining processes where very large capital investments are obligatory to obtain and replace the essential equipment. Accurate valuation of the system performance is not so easy as of the complexity of the system. Though, with some abridging conventions one can obtain equally accurate results using computer simulation methods for all applied purposes. One of the major matters in open-pit mining operations is the range of trucks and shovels that would please some financial and practical criteria optimally. This problem is confronted at the strategy stage of the mine as well as during the process of the mine where there may be a need to reshape for development purposes. The solution lies in effective calculation of performance factors for various groupings of trucks and shovels under accurate assumptions. These factors could be used to fix the effect of different situations on the output of the operation and select the best likely alternative for authentic design goals. Specified the characteristics of the truck-fleet, dynamic routing of trucks to dissimilar service areas (i.e. loading and dumping) cannot be done randomly since this would seriously affect the output of the mine. Therefore, it is very significant for optimal operation that the design factors should be determined precisely and applied at all phases of mining operation. Well-organized truck dispatching represents an out-dated approach to expand production equipment utilization in open-pit mining operations. Growing the equipment utilization can consequence in a greater increase in the lucrativeness of operation and reduction in the truck-fleet size as well as growth in production. Truck haulage signifies 50% or more of the total operating costs in most surface mines and labours have been made to decrease these high haulage costs. These comprise refining operating performance of the trucks bring about in higher efficiency and reliability, growing the payload capacity of trucks, engaging in-pit crushers and assigning systems with truck haulage, and with trolley-assisted trucks to decrease the truck cycle times. Another idea currently under progress is the use of driver-less trucks since this method has the prospective to lessen the labour costs. These hard works have concentrated on truck or haulage system designs. The same cost lessening goals can also be appreciated by more effective application of trucks and shovel resources, which is principal objective of computer-based truck dispatching systems. By computer-based truck dispatching, one expects either to proliferate manufacture with existing truck and shovel resources or meet the wanted manufacture goals with reduced equipment supplies. This goal is attained with careful thought of assignment decisions that grows use of truck and shovel capitals and lessen waiting times in the haulage network. Haulers are solitary creative when they are carrying a load and loaders are likewise only creative when loading material for haulage. Shiftless equipment times are the heart of non-productive equipment and they have to be reduced

To lessen risks of not meeting ore request due to operational qualms, mine operational staff often depend on providing extra trucks to haul ore material. This risk-averse method has led to inadequacy in truck habit, follow-on in long truck queues at dump locations, or at shovels, or at both. Allotting extra trucks to haul ore material describes that fewer trucks are vacant for other tasks such as hauling overburden, which can sometimes be acute during operation. Managers often decide short-term truck lacks by truck rents, but this resolve is costly.

Truck allocation and truck dispatching are two discrete procedures. Truck allocation is being emphasized, which characteristically takes place at the start of a shift. Truck dispatching repeatedly regulates the initial allocation based on the number of coming up trucks at shovels and dumps, equipment failure, and other features. The efficiency of the truck dispatching is greatly assisted by a good early allocation. Truck allocation models are also beneficial to managers, to govern or justify an acceptable budget for truck capitals, given the obligatory ore throughput.

Typically, truck dispatchers allocate trucks at the start of 8-hour shifts, based on past data and user skill. Effectiveness of truck allocation trusts on dispatchers' skill, which inclines to vary among shifts. Detailed truck-dispatching training is desirable to safeguard that dispatchers have an appropriate level of skill essential for truck allocation but such training does not assure steady or optimal truck allocation.

.

CHAPTER-2

Literature Review

2.1 Problem Statement

Suggesting and estimating an optimization-based truck allocation method, which faiths on queueing theory to express how average quantity is determined by the amount of trucks allocated to a shovel. The truck allocation model is stated via queueing theory where shovels are reproduced as servers with a finite number of trucks, as customers. Trucks are allocated to shovels to please manufacture constraints on amount and ore grade.

A key involvement of this paper is that, through the usage of queueing theory, we can capture the nonlinear connection between average throughput and the number of deployed trucks, and more prominently, integrate this connection in a tractable optimization model. This study validates that linearization is an actual method to integrate queueing theoretical relationships in the construction of a linear integer program. This technique permits analysts to integrate much more difficult constraints in a linear allocation model.

2.2 Queueing Theory

Queueing theory deals with the study of queues (waiting lines). Queues abound in practical situations. The earliest use of queueing theory was in the design of a telephone system. Applications of queueing theory are found in fields as seemingly diverse as traffic control, hospital management, and time-shared computer system design. In this chapter, we present an elementary queueing theory.

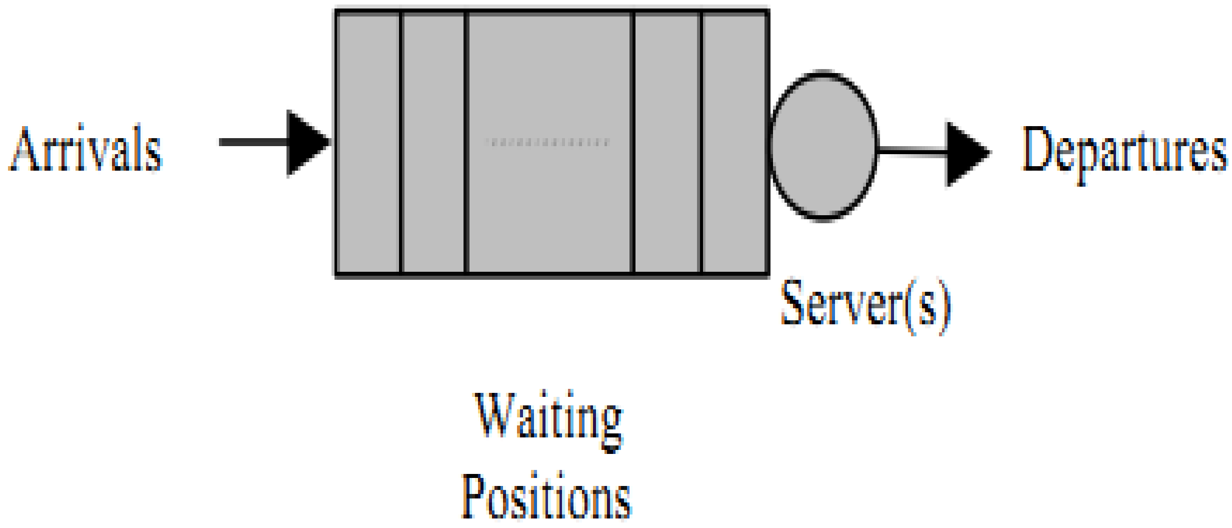

Figure 1: Model of a Queue

The following terms are commonly used in queueing theory.

Customers:

The persons or objects that require certain service are called customers.

Server:

The person or a machine that provides certain definite service is known as server.

Service:

The activity between server and customer is called service, this consumes some time.

Queue or Waiting Line:

A systematic arrangement of a group of persons or objects that wait for service.

Arrival:

The process of customers coming towards service facility or server to receive a certain facility.

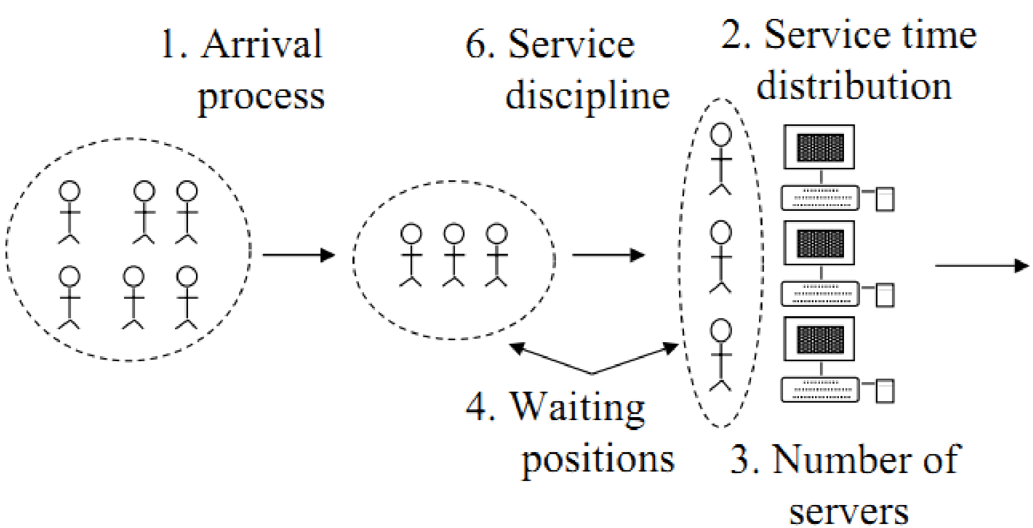

Figure 2: Queueing System

A queueing system can be completely described by

1. The input (Arrival pattern).
2. The service mechanism (Service pattern).
3. The queue discipline.
4. Customer's behaviour.

1. The input (Arrival pattern)

The input describes the way in which the customers arrive and join the system. Generally customers arrive neither in a more or less random fashion which is nor worth making the prediction. Thus the arrival pattern can be described in terms of probabilities and consequently the probability distribution for inter arrival times (the time between two successive arrivals) must be defined.

2. The service mechanism (Service pattern)

This means the arrangement facility to server customers. If there is infinite number of servers then all the customers are served instantaneously on arrival and there will be no queue.

If the number of servers is finite then the customers are served according to a specific order with service time a constant or a random variable. Distribution of service time which is important in practice is the negative exponential distribution. The mean service rate is denoted by μ.

3. The queue discipline

It is the rule according to which the customers are selected for service when a queue has been formed. The most common disciplines are:

- First come, first serve.
- First in, first out.
- Last in, first out.
- Selection for service in random order.

There are various other disciplines according to which the customer is served in preference over the other. Under priority discipline, the service is of two types, mainly pre-emptive and non-pre-emptive. In pre-emptive system, the high priority customers are given service over the low priority customers; in the non-pre-emptive system, a customer of low priority is serviced before a customer of high priority is entertained for service. In the case of parallel channels "*fastest server rule*" is adopted.

4. Customer's behaviour

The customers generally behave in the following four ways:

- Balking

  The customer who leaves the queue because the queue is too long and he has no time to wait or has no sufficient waiting space.

- Reneging

  This occurs when a waiting customer leaves the queue due to impatience.

- Priorities

  In certain application some customers are served before the others regardless of their arrival. These customers have priority over others.

- Jockeying

  Customers may jockey from one waiting line to another.

Many of these models further assume that all inter-arrival times are independent and identically dispersed and that all service times are in-dependent and identically distributed. Such models conventionally are labelled as follows:

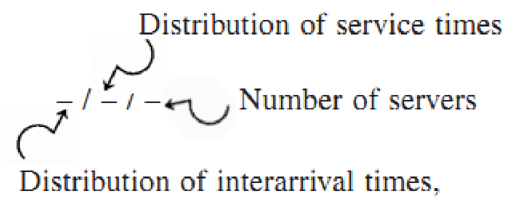

Figure 3: Kendall Notation

where

M = exponential distribution (Markovian),

D = degenerate distribution (constant times),

E_k = Erlang distribution (shape parameter = k),

G = general distribution (any arbitrary distribution allowed)

2.3 Linear Programming

Linear programming is used to solve optimisation problems taking linear objective function and constraints that is why these methods are used in operation research and transportation problem. Though linear programming methods are also used iteratively to answer non-linear programming problems where the objective function and/or constraints are non-linear.

2.3.1 Linear Programming Problem

Linear programming methods are used to solve problems having linear objective functions and linear constraints. Since all constraints are linear the feasible region is usually an enclosed region surrounded by linear hyper-planes. Since the objective function is also linear, the optimum point is usually one of the corner points of the feasible region.

There are two dissimilar ways one can get a feasible solution of the unique problem. One way to attain this is to select the set of basic variables a priori, allot several sets of values to non-basic variables, and then acquire agreeing basic variables in each case. The other way is to select more than a few sets of basic variables, allot all non-basic variables to zero, and then calculate the values of the basic variables in each circumstance. In this technique, the latter method is used. In this method, any mixture of J variables can be selected to create the row-echelon formulation. Since there are N variables from which to select J variables, there are total of $\binom{N}{J}$ dissimilar solutions to relate. A basic feasible solution is defined as a solution at which all basic variables are non-negative and all non-basic variables are zero. Thus there is total number of $\binom{N}{J}$ basic feasible solutions likely, of which one or more agrees to the optimum point. A simple approach would be to match all such basic feasible solutions and find the optimum. An effective approach would be to link only a small fraction of all basic feasible solutions to get the optimum.

2.3.2 Simplex Method

The simplex method was technologically advanced by G. B. Dantzig in 1963. The main idea of this simplex method is to equate neighbouring basic feasible solutions in an effective way. A neighbouring solution is a basic feasible solution which varies from the current basic feasible solution in precisely on basic variable. That is, one basic variable in the existing basic feasible solution is made non-basic and a presently non-basic variable is made basic. This can be attained in a total of J(N-J) dissimilar ways. Therefore, there are two choices to be made:

1. One of the (N-J) non-basic variables is to be selected for the basic variable.
2. One of the J basic variables is to be selected for the non-basic variable.

It is clear that the superior of non-basic variables will outcome in different objective function values. The simplex method selects that non-basic variable which effects in a maximum rise in the objective function value. By simple algebraic scheming, it can be revealed that the rise in the objective function value *f(x)* due to a unit rise in a non-basic variable x_q from zero to one is given by **[2]**

$$(\Delta f)q = c_q - \sum_{j=1}^{j} c_j a'_{jq}$$

Thus, we can compute $(\Delta f)q$ for all non-basic variables and select the one with the maximum positive value. In the case of a equal finish (similar $(\Delta f)q$ for other than one non-basic variables), any non-basic variable can be selected at casual. It is worth stating here that if for all residual non-basic variables, the quantity $(\Delta f)q$ is non-positive, no increase in the objective function value is conceivable. This proposes that the optimum solution has been found, and we terminate the algorithm.

Once a non-basic variable (x_q) is selected, the next question is, which of the basic variables has to be made non-basic. It is clear that as x_q is increased from zero to one, the objective function value will also rise. Consequently we may want to raise the non-basic variable x_q indeterminately. But there is a boundary to the extent of this increase, when x_q is increased, all basic variables must be increased, decreased, or kept similar in order to create the solution feasible. Variables in a linear program essential be non-negative. Thus, the critical basic variable is the one which, when decreased, becomes zero first. Any more increase in the chosen non-basic variable will make that basic variable negative. After the row-echelon formulation, we can write the value of a basic variable as follows **[2]**. $x_j = b'_j - a'_{jq} x_q\ , j = 1,2, \dots ., J.$

A basic variable becomes zero when $x_q = b'_j / a'_{jq}$. Since, in the row-echelon form all b'_j are non-negative, this can occur only when a'_{jq} is positive. In order to catch the critical basic variable, we calculate the quantity b'_j / a'_{jq} and select the basic variable x_q for which this quantity is less. This rule is also known as *minimum ratio* rule in linear programming. The simplex method initiates with an initial feasible solution. Thereafter, a basic variable is swapped by a non-basic variable selected according to rules defined. Thus, the simplex method is an iterative process which works by interchanging among various basic and non-basic variables so as to attain the optimum point capably. Since all constraints and the objective function are linear, these points are the corners of the feasible search region. In the following we describe the algorithm:

Step 1

A basic feasible solution is chosen. All non-basic variables are set to zero.

Step 2

The quantity $(\Delta f)q$ is calculated for all non-basic variables and the one having the maximum value is chosen. If $(\Delta f)q$ <= 0, for all non-basic variables, *Terminate*:

Else the minimum ratio rule is used to choose the basic variable to be replaced.

Step 3

Performing a row-echelon formulation for new basic variables and then going to *step 2.*

2.3.3 Artificial variables and dual phase method

The working technique of this simplex method is straight-forward and is easy to apply. There are some commercial codes applying the simplex method (IBM-MPSX-370 or Management science systems MPS-III). Though a successful LP code must handle a complexity usually encountered in many LP problems. The complexity is that in some LP problems it may not be likely to obtain an early basic feasible solution straight from the given constraints. In those circumstances, *artificial* variables are added. Artificial variables have no sense as far as the difficulty is concerned. They are simply involved to obtain a basic feasible solution. Once a basic feasible solution is obtained, the artificial variables can be omitted from further deliberation. Typically a dual phase strategy is used for this process. In the first phase, the simplex method is used to find a basic feasible solution, including the design variables, slack variables and artificial variables. The objective of the exploration strategy is to maximise the negative of the sum of the artificial variables. Then the artificial variables are also constrained to be non-negative, the optimal solution of the first phase is a solution on which all artificial variables are zero. In the meantime at the end of the first phase all artificial variables are zero, the basic variables are any design variables or slack variables. This establishes a basic feasible solution which can be used as the initial solution for the next phase, where *f(x)* is used as the objective function.

CHAPTER-3

Allocation Model

3.1 Truck Allocation Model

The aim of this project is to formulate the task of allocating a minimum total number of equal-sized trucks to a set of shovels (S) to content a lower limit (L) on throughput and lower and upper limits, g_l and g_u, on ore grade. The function of the number of trucks allocated to a particular shovel is shovel throughput, which can be linearized, that results in a linear integer program which is verified here.

3.1.1 Shovel Throughput

The operation of t trucks is allocated to a shovel as a finite source $G/G/1/\text{-}/t$ queueing system. The shovel (or server) is characterized by the shovel service time distribution with mean 1/µ and squared coefficient of variation s_s^2(variance / mean^2). The trucks (the customers) are characterized by a back-cycle time distribution with mean 1/ƛ, and squared coefficient of variation s_a^2 .The back-cycle time is the truck cycle time minus shovel waiting and service times. Mean service time depends on both shovel capacity and truck capacity. Mean back-cycle time depends on travel distance, truck speed, and dump capacity. The steady state shovel throughput (W) depends on the average service rate while busy (μ) the average truck capacity (C), the number of allocated trucks (t) and the shovel idle probability, $P_0(t)$ **[6]**:

$$W = \mu\,[1 - P_0(t)]C.$$

It is predictable that the shovel idle probability to lessen and throughput to rise with the fleet size (t) irrespective of the shapes of the service time and back-cycle time distributions. Fig. 3 demonstrates how the idle probability reduces with fleet size in $G/M/1/\text{-}/t$ and $D/D/1/\text{-}/t$

systems, with 1/ ƛ = 14 minutes and 1/µ = 26 minutes. The idle probability for these two models can be defined as a function of $\rho = ƛ/\mu$ and t in closed form **[6]**, as follows:

$$P_0^{G/M/1}\ (t) = \frac{\rho^{-t}/t!}{\sum_{k=0}^{t} \rho^{-k}/_{k!}}, \qquad (1)$$

$$P_0^{D/D/1}\ (t) = \begin{cases} 1 - \dfrac{t}{1+1/\rho} & t \le 1+1/\rho \\ 0 & t > 1+1/\rho \end{cases}, \qquad (2)$$

The idle probability $P_0^{G/M/1}(t)$ in a *G/M/1/-/t* queueing model, with exponentially distributed service times is unresponsive to the shape of the back-cycle time distribution outside its mean. The steady state characteristics are nearly unresponsive to the shape of the back-cycle time distribution even if the service times are not exponentially distributed. This inspires the following approximation **[6]**

$$P_0^{G/G/1}\ (t) = \left[\frac{1+s_s^2}{2}\right] P_0^{G/M/1}\ (t) + \left[1 - \frac{1+s_s^2}{2}\right] P_0^{D/D/1}\ (t), (3)$$

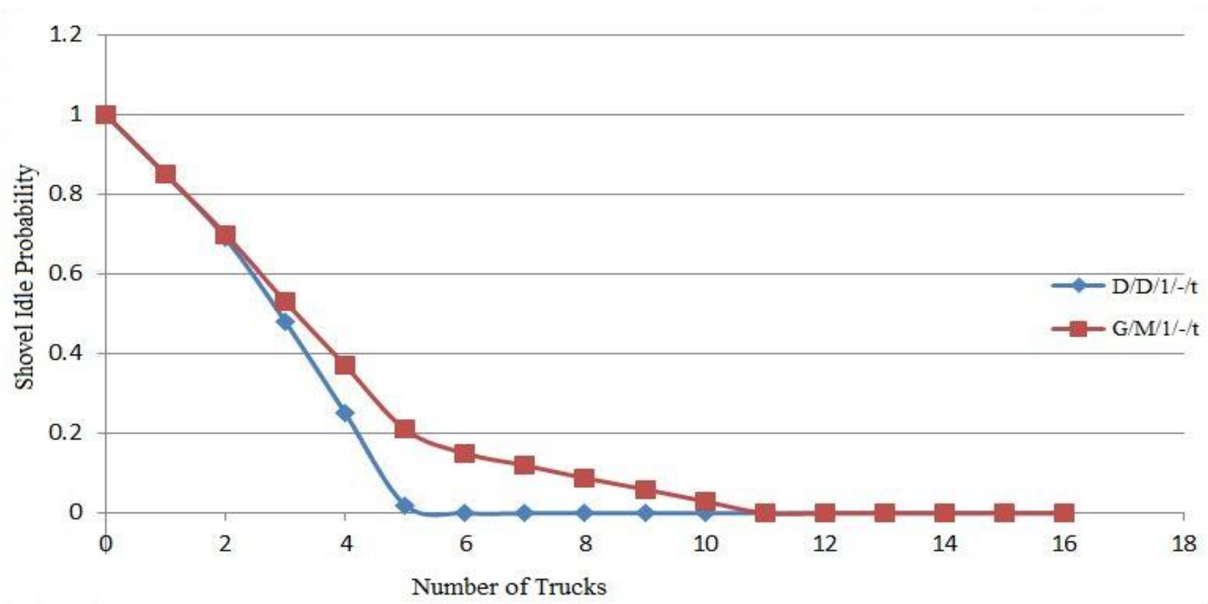

Figure 4: Idle probability functions for G/M/1-/n and D/D/1-/n systems

Table 1: Idle probability values for G/M/1-/n and D/D/1-/n systems

Number of Trucks	$P_0^{D/D/1}\ (t)$	$P_0^{G/M/1}\ (t)$
0	1	1
1	0.85	0.85
2	0.69	0.70
3	0.48	0.53
4	0.25	0.37
5	0.02	0.21
6	0	0.15
7	0	0.12
8	0	0.088
9	0	0.06
10	0	0.03
11	0	0
12	0	0
13	0	0
14	0	0
15	0	0
16	0	0

Fig. 4 was generated by using equation (1) and (2) and Table 1. The idle probability graph of D/D/1/-/t model with no. of trucks ceases to zero at 5.5 number of trucks and that of G/M/1/-/t decreases gradually after 5.5 number of trucks, which implies that if we increase the number of trucks more than 6 trucks will be in queue and shovel will work without being idle.

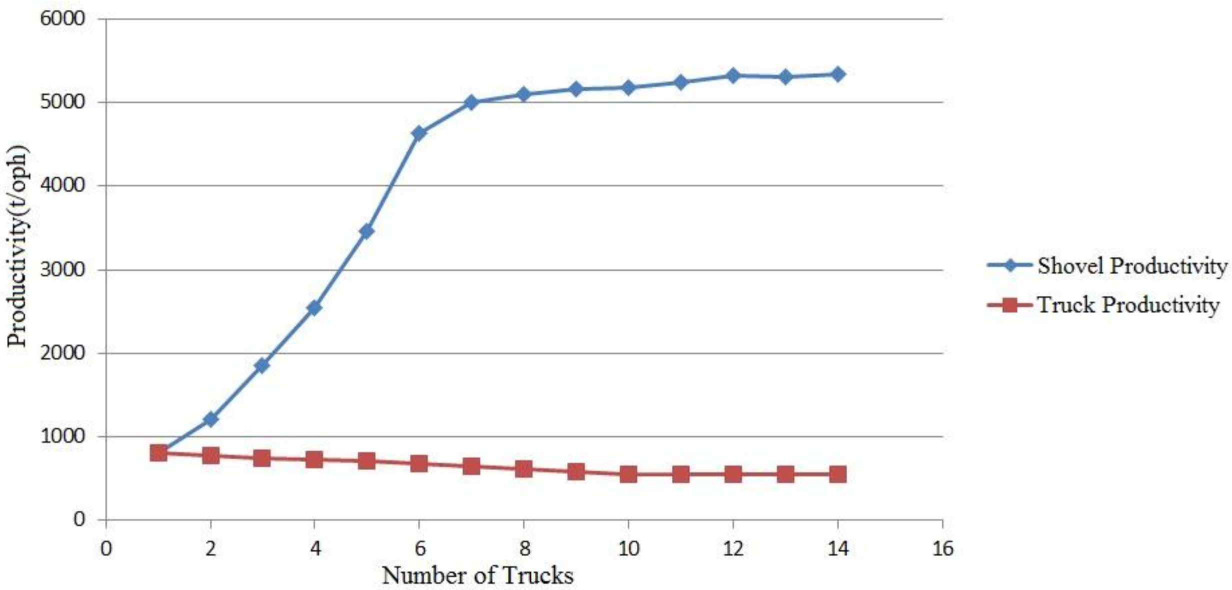

Figure 5: Productivity versus Fleet size

Table 2: Shovel Productivity and Truck Productivity Data

Number of trucks	Shovel Productivity	Truck Productivity
1	803	805
2	1203	770
3	1847	750
4	2551	733
5	3458	713
6	4634	687
7	5011	653
8	5101	611
9	5157	583

10	5173	551
11	5252	551
12	5321	551
13	5311	548
14	5336	547

Fig 5 was generated using the production data from *NALCO damanjodi* listed in Table 2. Fig 5 shows shovel productivity increases and truck productivity decreases with fleet size. It can be seen that at 5-6 number of trucks the graph of shovel continues to increase gradually and the increase is not significant in Productivity, and it implies that after allocation 6 trucks productivity does not increase significantly. The graph of truck decreases rapidly after 6 number of trucks, by decreasing the productivity.

3.1.2 Linearization of Idle Probability Function

Linearizing the idle probability as a function of the number of allocated trucks as follows **[6]**:

For each shovel:

$$\rho_{0(z)} = \sum_{k=0}^{m} w_k x_k, \tag{4}$$

$$\sum_{k=0}^{m} w_k x_k = 1, \tag{5}$$

$$z = \sum_{k=0}^{m} k x_k, \tag{6}$$

z is the number of trucks and x_k is a binary variable for the condition $k = z$. Constraints (5) and (6) ensure that variables z and x_k take steady values, causing shovel throughputs being stated as a linear function of the variables just stated **[6]**:

$$W(z) = \mu\left[1 - \sum_{k=0}^{m} w_k x_k\right] C \qquad (7)$$

Now, we use the expression for throughput to formulate our basic allocation model, which assumes that all trucks have the same capacity.

3.1.3 Linear Programming

Our basic integer program allocates trucks to shovels so as to minimize the total number of trucks, assuming that all trucks have the same capacity. The model has three sets of decision variables: binary linearization variables (x_{kj}), integer variables for the number of trucks allocated to shovel j (z_j), and continuous variables for the throughput of shovel j (Wj).

We denote S as a set of shovels and $I = \{0,\dots,m\}$ as a set of truck indices. Parameters specific to shovel j include service rates (μ_j), ore grades (g_j) and average truck capacity (C) . The parameters w_{kj} depend on the loading capacity of shovel j. Their values depend on both shovel service times and truck back-cycle times. Ore demand L and an ore grade range, $[g_l, g_u]$, complete the required input parameters. **[6]**

Basic: minimize

$$\sum_{j \epsilon S} z_j$$

Subject to

$$\sum_{j \epsilon S} W_j \geq L, \tag{8}$$

$$\sum_{j \epsilon S} g_j \, W_j \leq g_u \sum_{j \epsilon S} W_j, \tag{9}$$

$$\sum_{j \epsilon S} g_j \, W_j \geq g_l \sum_{j \epsilon S} W_j, \tag{10}$$

$$W_j = \mu_j \left(1 - \sum_{k=0}^{m} w_{kj} x_{kj} \right) C, \quad \forall j \epsilon S \tag{11}$$

$$z_j = \sum_{k \epsilon I} k x_{kj}, \tag{12}$$

$$\sum_{k \epsilon I} x_{kj} = 1, \tag{13}$$

$$x_{kj} \epsilon \, \{0,1\}, \tag{14}$$

$$W_j \geq 0, z_j \geq 0, \tag{15}$$

Constraint (8) ensures that a sufficient number of trucks are allocated to meet the ore demand. Constraints (9) and (10) guarantee that the ore grade is within the specified range. Equality constraints (11) define the shovel throughput while constraints (12) and (13) relate the binary variables, x_{kj}, to the number of allocated trucks, z_j. Note that the integer variables z_j can be confirmed as continuous, because constraints (12) and (14) ensure that z_j takes integer values. The above model is based on the following assumptions:

- Each truck is assigned to the same shovel for the duration of the time shift.
- All trucks are of the same size.

- Ore mixing is ideal at the dump locations, resulting in a consistent calculation of the combined ore grade.

We relax the assumption of a single truck size in the next section. The first two assumptions are important for our methodology, as they permit decomposition of the system by shovel. Queueing network approximations or simulation can be used to evaluate the performance of truck-shovel systems where the first two assumptions are violated, but in such situations, the idle probability of a shovel will depend not only on the number of trucks assigned to that shovel but also on the number of trucks assigned to other shovels, as well as the capacity of, for example, dump locations and repair facilities. Therefore, if the first two assumptions are violated, it is not possible to view the parameters w_{ij} as input parameters for the optimization problem—rather; they become quantities whose values depend on the decision variables.

CHAPTER-4

Conclusions

4.1 Conclusion

It has been established that server throughput calculated using queueing theory and linear programming can be efficiently applied in allocating trucks to shovels. By using a linearized throughput function, simple linear integer programs can be formulated. And different optimization programs/software can be used to maximize the shovel throughput. By minimizing the number of trucks allocated to a shovel; no fleet being idle, production of the mine can be increased. This technique can be used to frame further allocation models with complex constraints using queueing theory. As compared to other formulation methods, this technique allows to calculate shovel performance (e.g., shovel idle probabilities), which is related with the optimal truck allocation.

References:

1. Taha, H.A, Operations Research-An Introduction, 8th Edition, New Delhi: Prentice Hall of India Pvt. Ltd., 2008.
2. Deb, K, Optimization for Engineering Design-Algorithms and Examples, New Delhi: PHI Learning Pvt. Ltd., 2009.
3. Gross, D, Harris, C.M, Fundamentals of Queueing Theory, 3rd Edition, Singapore: John Wiley & Sons, Inc, 2004.
4. Singh, L.K, R.M.L, Srivastava, R, "Estimation of Buffer Size of Internet Gateway Server via G/M/1 Queuing Model", World Academy of Science, Engineering and Technology, Vol33, 2007, pp-98-106.
5. www2.mining.unsw.edu.au/Publications/publications_staff/Paper_NajorHagan_Capacity Production.htm. (2012, 03 15). Retrieved from www2.mining.unsw.edu.au.com.
6. Ta, H.T, Ingolfsson, A, Doucette, J, "Haul Truck Allocation via Queueing Theory", European Journal of Operational Research, October 2010, pp-1-20.
7. http://www.cse.wustl.edu/~jain/cse567-08/ftp/k_30iqt/index.htm. (2012, 03 20). Retrieved from http://www.cse.wustl.edu.
8. http://www.win.tue.nl/~iadan/queueing.pdf. (2012, 03 20). Retrieved from http://www.win.tue.nl.
9. http://home.iitk.ac.in/~skb/ee679/ee679.html. (2012. 04 20). Retrieved from http://home.iitk.ac.in/~skb/qbook/Slide_Set_1.PDF.